

**IN THE UNITED STATES PATENT AND TRADEMARK OFFICE  
BEFORE THE BOARD OF PATENT APPEALS AND INTERFERENCES**

In re Application of:  
Richard D. Dettinger et al.

Serial No.: 10/601,995

Confirmation No.: 9828

§  
§  
§  
§  
§

Filed: June 23, 2003

Group Art Unit: 3628

Examiner: Allen J. Jung

For: METHOD OF ESTABLISHING A DATA MANAGEMENT FEE STRUCTURE  
BASED ON FINE GRAINED DATA ENTITIES

MAIL STOP APPEAL BRIEF - PATENTS  
Commissioner for Patents  
P.O. Box 1450  
Alexandria, VA 22313-1450

**CERTIFICATE OF MAILING OR TRANSMISSION**

I hereby certify that this correspondence is being deposited with the United States Postal Service with sufficient postage as first class mail in an envelope addressed to: Mail Stop Appeal Brief - Patents, Commissioner for Patents, P. O. Box 1450, Alexandria, VA 22313-1450, or facsimile transmitted to the U.S. Patent and Trademark Office to fax number 571-273-8300 to the attention of Examiner Allen J. Jung, or electronically transmitted via EFS-Web, on the date shown below:

December 16, 2009  
Date

/Johnny Lam/  
Johnny Lam

Dear Sir:

**APPEAL BRIEF**

Applicants submit this Appeal Brief to the Board of Patent Appeals and Interferences on appeal from the decision of the Examiner of Group Art Unit 3628 dated June 17, 2009, finally rejecting claims 1-37, 46-53 and 55-74. The final rejection of claims 1-37, 46-53 and 55-74 is appealed. This Appeal Brief is believed to be timely since it is transmitted by the due date of December 16, 2009, as set by the filing of a Notice of Appeal on October 16, 2009.

Please charge the fee of \$540.00 for filing this brief to:

Deposit Account No. 09-0465 / ROC920030158US1.

**TABLE OF CONTENTS**

1.	Identification Page.....	1
2.	Table of Contents .....	2
3.	Real Party in Interest .....	3
4.	Related Appeals and Interferences .....	4
5.	Status of Claims .....	5
6.	Status of Amendments .....	6
7.	Summary of Claimed Subject Matter .....	7
8.	Grounds of Rejection to be Reviewed on Appeal .....	13
9.	Arguments .....	14
10.	Conclusion .....	20
11.	Claims Appendix .....	21
12.	Evidence Appendix .....	36
13.	Related Proceedings Appendix .....	37

### **Real Party in Interest**

The present application has been assigned to International Business Machines Corporation, Armonk, New York.

### **Related Appeals and Interferences**

Applicant asserts that no other appeals or interferences are known to the Applicant, the Applicant's legal representative, or assignee which will directly affect or be directly affected by or have a bearing on the Board's decision in the pending appeal.

### **Status of Claims**

Claims 1-37, 46-53 and 55-74 are pending in the application. Claims 1-74 were originally presented in the application. Claims 38-45 and 54 have been canceled without prejudice. Claims 1-37, 46-53 and 55-74 stand finally rejected as discussed below. The final rejections of claims 1-37, 46-53 and 55-74 are appealed. The pending claims are shown in the attached Claims Appendix.

### **Status of Amendments**

All claim amendments have been entered by the Examiner, including amendments to the claims proposed after the final rejection.

## **Summary of Claimed Subject Matter**

### **A. CLAIM 1 – INDEPENDENT**

Claim 1 recites a computer implemented method of providing fee-based access to data. See Application, p. 11, lines 1-5. The method includes providing an abstract model for logically defining abstract operations to access the data. See Application, p. 11, lines 6-18. The abstract model includes a plurality of logical fields. See Application, Figure 2B, element 208; p. 11, lines 19-28. The abstract model also includes a mapping rule for each of the plurality of logical fields, which map the plurality of logical fields to physical entities of the data, and wherein each mapping rule comprises an access method which is executed to retrieve a respective physical field of the physical entities of the data. See Application, p. 19, lines 2-25. The abstract model also includes a fee schedule for each of the plurality of logical fields, wherein each fee schedule for a given logical field defines a fee to be charged when the given logical field is involved in an abstract operation to access a physical entity corresponding to the given logical field. See Application, Figure 31, element 530A; p. 51, line 20 – p. 52, line 14. The method also includes executing a fee calculator on a computer processor to calculate, based on the fee schedules, a fee to be charged for accessing the data. See Application, Figure 1, element 151; p. 53, line 5 – p. 56, line 14.

### **B. CLAIM 8 - INDEPENDENT**

Claim 8 recites a computer implemented method of providing fee-based access to physical data comprising a plurality of physical entities each comprising a plurality of physical fields. See Application, p. 11, lines 1-5. The method includes providing an abstract model for defining abstract operation specifications logically describing operations to access the data. See Application, p. 11, lines 6-18. The abstract model includes a plurality of logical fields. See Application, Figure 2B, element 208; p. 11, lines 19-28. The abstract model also includes a mapping rule for each of the plurality of

logical fields, which map each of the plurality of logical fields to at least one of the physical entities of the data, and wherein each mapping rule comprises an access method which is executed to retrieve a respective physical field of the physical entities of the data. See Application, p. 19, lines 2-25. The abstract model also includes a plurality of model entity definitions, each comprising at least one logical field corresponding to a physical field of a physical entity. See Application, Figure 2B, element 208; p. 11, lines 19-28. The abstract model also includes a logical field fee schedule for each of the plurality of logical fields, wherein the fee schedules each specify a fee for accessing a corresponding physical field as part of a physical operation. See Application, Figure 31, element 530A; p. 51, line 20 – p. 52, line 14. The method also includes executing a fee calculator on a computer processor to calculate, based on the fee schedules, a fee to be charged for accessing the data. See Application, Figure 1, element 151; p. 53, line 5 – p. 56, line 14.

#### C. CLAIM 12 - INDEPENDENT

Claim 12 recites a computer implemented method of providing fee-based access to data comprising a plurality of physical entities, each including a plurality of physical fields. See Application, p. 11, lines 1-5. The method includes receiving instructions to perform an operation for accessing the data. See Application, p. 53, line 16 – p. 55, line 4. The method also includes performing the operation. See Application, p. 53, line 16 – p. 55, line 4. The method also includes determining field-specific fees for each of a plurality of the physical fields accessed by the operation. See Application, Figure 31, element 530A; p. 51, line 20 – p. 52, line 14. The method also includes calculating, by operation of a computer processor, a total fee to be charged to a user for the operation. See Application, Figure 1, element 151; p. 53, line 5 – p. 56, line 14.



D. CLAIM 29 - INDEPENDENT

Claim 29 recites a computer-readable storage medium containing a program which, when executed by a processor, performs operations for accessing physical data comprising a plurality of physical entities, each having a plurality of physical fields. See Application, p. 11, lines 1-5; p. 12, lines 5-17. The operations include receiving instructions to perform an operation accessing the data. See Application, p. 53, line 16 – p. 55, line 4. The operations also include causing performance of the operation. See Application, p. 53, line 16 – p. 55, line 4. The operations also include determining field-specific fees for each of a plurality of the physical fields accessed by the operation. See Application, Figure 31, element 530A; p. 51, line 20 – p. 52, line 14. The operations also include calculating a total fee to be charged to a user for the operation. See Application, Figure 1, element 151; p. 53, line 5 – p. 56, line 14.

E. CLAIM 46 - INDEPENDENT

Claim 46 recites a computer implemented method of providing a logical framework for defining abstract operations for accessing physical data including a plurality of physical entities each comprising a plurality of physical fields. See Application, p. 11, lines 1-5. The method includes providing an abstract model for defining abstract operation specifications logically describing operations to access the data. See Application, p. 11, lines 6-18. The abstract model includes a plurality of logical fields. See Application, Figure 2B, element 208; p. 11, lines 19-28. The abstract model also includes a mapping rule for each of the plurality of logical fields, which map each of the plurality of logical fields to at least one of the physical entities of the data and wherein each mapping rule includes an access method which is executed to retrieve a respective physical field of the physical entities of the data. See Application, p. 19, lines 2-25. The abstract model also includes a plurality of model entity definitions, each including at least one logical field corresponding to a physical field of a physical entity. See Application, Figure 2B, element 208; p. 11, lines 19-28. The abstract model

also includes model entity fee schedules for each of the plurality of model entity definitions, wherein the fee schedules each specify a fee for accessing a physical field of the corresponding model entity definition. See Application, Figure 31, element 530A; p. 51, line 20 – p. 52, line 14. The method also includes providing a run-time component to transform, according to the abstract model, abstract operation specifications into physical operation specifications consistent with the physical data, wherein each abstract operation specification includes at least one user-selected model entity definitions of the plurality of model entity definitions. See Application, page 6, line 24 – page 7, line 18. The method also includes executing a fee calculator on a computer processor to calculate, based on the fee schedules, a fee to be charged for accessing the data. See Application, Figure 1, element 151; p. 53, line 5 – p. 56, line 14.

#### F. CLAIM 59 - INDEPENDENT

Claim 59 recites a computer-readable storage medium containing a program which, when executed by a processor, provides a logical framework for defining abstract query operations. See Application, p. 11, lines 1-5; p. 12, lines 5-17. The program includes an abstract model for defining abstract queries logically describing operations to query the data. See Application, p. 11, lines 6-18. The abstract model includes a plurality of logical fields. See Application, Figure 2B, element 208; p. 11, lines 19-28. The abstract model also includes a mapping rule for each of the plurality of logical fields, which map the plurality of logical fields to physical entities of the data, wherein each mapping rule includes an access method which is executed to retrieve a respective physical field of the physical entities of the data. See Application, p. 19, lines 2-25. The abstract model also includes a fee schedule for each of the plurality of logical fields. See Application, Figure 31, element 530A; p. 51, line 20 – p. 52, line 14. The program also includes a run-time component configured with transformation instructions to transform an abstract query, which includes logical fields selected from the plurality of logical fields, into a physical query consistent with the physical data. See Application,

page 6, line 24 – page 7, line 18. The program also includes a fee calculator configured to calculate a fee for executing physical queries based on the fee schedules. See Application, Figure 1, element 151; p. 53, line 5 – p. 56, line 14.

#### G. CLAIM 61 - INDEPENDENT

Claim 61 recites a computer that includes a memory, at least one computer processor, and a logical framework for defining abstract modification operations for modifying physical data. See Application, p. 7, lines 7-18; p. 11, lines 1-5. The logical framework includes an abstract model for defining an abstract modification specification logically describing an operation to modify the data. See Application, p. 11, lines 6-18. The abstract model includes a plurality of logical fields. See Application, Figure 2B, element 208; p. 11, lines 19-28. The abstract model also includes a mapping rule for each of the plurality of logical fields, which map the plurality of logical fields to physical entities of the data, wherein each mapping rule comprises an access method which is executed to retrieve a respective physical field of the physical entities of the data. See Application, p. 19, lines 2-25. The abstract model also includes a fee schedule for each of the plurality of logical fields. See Application, Figure 31, element 530A; p. 51, line 20 – p. 52, line 14. The logical framework also includes a run-time component to transform an abstract query, which includes logical fields selected from the plurality of logical fields, into a physical query consistent with the physical data. See Application, page 6, line 24 – page 7, line 18. The logical framework also includes a fee calculator which when executed configures the at least one computer processor to calculate a fee to be charged for executing physical queries based on the fee schedules. See Application, Figure 1, element 151; p. 53, line 5 – p. 56, line 14.

#### H. CLAIM 64 - INDEPENDENT

Claim 64 recites a computer implemented method of providing fee-based access to data that includes a plurality of physical entities, each including a plurality of physical fields. See Application, p. 11, lines 1-5. The method includes receiving, via a user interface, user input comprising instructions for an operation for accessing the data selected fields of the plurality of the physical fields. See Application, p. 53, line 16 – p. 55, line 4. The method also includes determining field-specific fees for each of the selected fields. See Application, Figure 31, element 530A; p. 51, line 20 – p. 52, line 14. The method also includes calculating, by a computer processor, a fee to be charged to a user for accessing the selected fields. See Application, Figure 1, element 151; p. 53, line 5 – p. 56, line 14. The method also includes displaying the fee to the user via a user interface. See Application, p. 56, line 19 – p. 57, line 28.

#### I. CLAIM 70 - INDEPENDENT

Claim 70 recites a computer implemented method for displaying fee information for fee-based access to data that includes a plurality of physical entities, each including a plurality of physical fields. See Application, p. 11, lines 1-5. The method includes displaying one or more user interface screens for construction of queries. See Application, p. 7, line 26 – p. 8, line 4. The method also includes receiving, via the one or more user interface screens, user input defining a query configured to access selected fields of the plurality of physical fields. See Application, p. 7, line 26 – p. 8, line 4. The method also includes displaying, via the one or more user interface screens on an output device, a field-specific access fee for each of the selected fields. See Application, p. 51, line 20 – p. 52, line 14; p. 53, line 5 – p. 56, line 14.

### **Grounds of Rejection to be Reviewed on Appeal**

1. Rejection of claims 70-74 under 35 U.S.C.101 because the claimed invention is directed to non-statutory subject matter.
2. Rejection of claim 61 under 35 U.S.C. 101 because the claimed invention is directed to nonstatutory subject matter.
3. Rejection of claims 1-37, 46-50 and 55-74 under 35 U.S.C. 103(a) as being unpatentable over *Coutts et al* (US 2002/0073066 A1) (hereinafter *Coutts*), in view of *Rao et al* (US 2003/0110087 A1) (hereinafter *Rao*).
4. Rejection of claims 51-53 under 35 U.S.C. 103(a) as being unpatentable over *Coutts*, in view of *Rao*, and further in view of *Lyons et al* (US 4,989,141) (hereinafter *Lyons*).

## **ARGUMENTS**

### **1. Rejection of claims 70-74 under 35 U.S.C.101 because the claimed invention is directed to non-statutory subject matter.**

#### *The Applicable Law*

The current standard for determining statutory subject matter is set forth in *In re Bilski*, 545 F.3d 943 (Fed. Cir. 2008) (en banc). The standard of *In re Bilski* is commonly referred to as the “machine or transformation” test:

“A claimed process is surely patent-eligible under § 101 if: (1) it is tied to a particular machine or apparatus, or (2) it transforms a particular article into a different state or thing.”

*Id.* at 954.

#### *Applicants’ Response to the Examiner’s Argument*

Claim 70 recites “displaying, via the one or more user interface screens on an output device, a field-specific access fee for each of the selected fields.” That is, the claim recites the apparatus (i.e., the output device) on which the field-specific access fee is displayed. Therefore, claim 70 is tied to a particular machine or apparatus – namely, the output device – and is therefore directed to statutory subject matter. Accordingly, Applicants respectfully submit that the rejection is defective and should be reversed with respect to claim 70.

### **2. Rejection of claim 61 under 35 U.S.C. 101 because the claimed invention is directed to nonstatutory subject matter.**

#### *Applicants’ Response to the Examiner’s Argument*

Claim 61 recites “executing a fee calculator on a computer processor to calculate, based on the fee schedules, a fee to be charged . . . .” That is, the claim recites the apparatus (i.e., the processor) that executes a fee calculator to calculate a

fee to be charged. Therefore, claim 61 is tied to a particular machine or apparatus – namely, the computer processor – and is therefore directed to statutory subject matter. Accordingly, Applicants respectfully submit that the rejection is defective and should be reversed with respect to claim 61.

**3. Rejection of claims 1-37, 46-50 and 55-74 under 35 U.S.C. 103(a) as being unpatentable over *Coutts* in view of *Rao*.**

*The Applicable Law*

The Examiner bears the initial burden of establishing a prima facie case of obviousness. See MPEP § 2141. Establishing a prima facie case of obviousness begins with first resolving the factual inquiries of *Graham v. John Deere Co.* 383 U.S. 1 (1966). The factual inquiries are as follows:

- (A) determining the scope and content of the prior art;
- (B) ascertaining the differences between the claimed invention and the prior art;
- (C) resolving the level of ordinary skill in the art; and
- (D) considering any objective indicia of nonobviousness.

Once the *Graham* factual inquiries are resolved, the Examiner must determine whether the claimed invention would have been obvious to one of ordinary skill in the art.

*Applicants' Response to the Examiner's Argument*

Respectfully, Applicants submit that the Examiner has not properly characterized the teachings of the references and/or the claims at issue. Accordingly, a *prima facie* case of obviousness has not been established.

For example, the Examiner suggests that *Coutts*, in view of *Rao*, teaches “each fee schedule for a given logical field defines a fee to be charged when the given logical field is involved in an abstract operation to access a physical entity corresponding to the given logical field” as recited in claim 1. Independent claims 8, 12, 29, 46, 59, 61, 64 and 70 recite similar limitations. Specifically, the Examiner asserts as follows:

*Coutts* does not explicitly disclose that such fee schedule is set up in such a way that “each fee schedule for a given logical field defines a fee to be

charged.” However, Rao discloses that “pricing structures can include different prices for access to different types of data,” clearly teaching that as a subscriber accesses different data of different types, each type of data would incur its own corresponding fee in order to form the overall price to be charged to the subscriber. It would have been obvious to one of ordinary skill in the art at the time of invention to combine Coutts’ data access service, with the pricing structure as taught by Rao.

Final Office Action, pages 9-10 (emphasis original). However, *Coutts*, even in view of *Rao*, teaches no such limitation. To illustrate, *Rao* describes “data types” as follows:

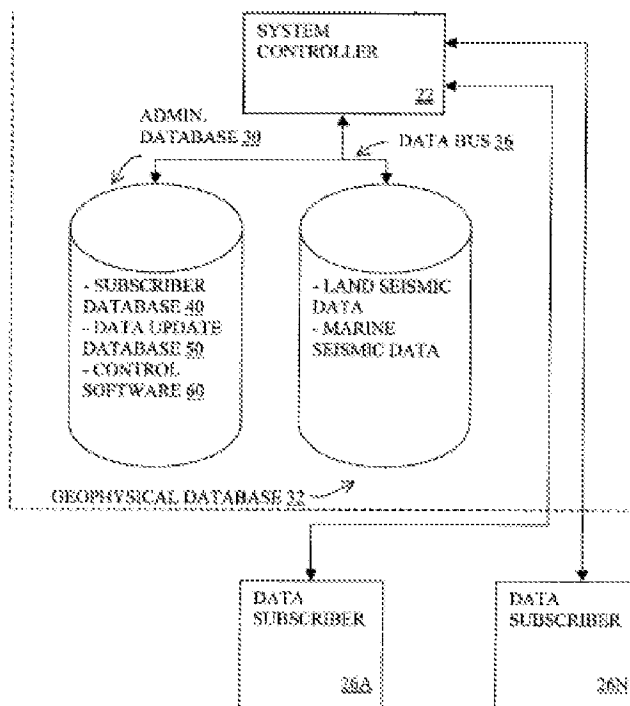


Fig. 1

Geophysical database 32, as described in further detail below, can contain many different types of data including both land and marine seismic data .

...

*Rao*, Fig. 1; ¶ 47 (emphasis added). That is, *Rao* describes a geophysical database 32 that stores different types of data, such as land seismic data and marine seismic data. *Rao* further describes data types as follows:



	34A	34B	34C	34D	34E	34F	34G	34H
	DATA SHOOT ID	DATA DESCRIPTION	DATA PROVIDER & TERMS	COLLECTION DATE	RELATED DATA LINK	DATA USAGE INFORMATION	DATA QUALITY INFORMATION	DATA
33A	GC1	GULF COAST MARINE AREA ABC; 1000 SQUARE MILES; 3D	ABC COLLECTOR; LICENSED UNDER XYZ TERMS & CONDITIONS	01/01/1998-03/01/1998	LINK TO UPDATE SHOOT	(USAGE INFORMATION BY USER AND DATA UNIT)	QUALITY INFORMATION	(LINK TO DATA SET)
33B	ME3	MIDDLE EAST LAND AREA; 1500 SQUARE MILES; 4D	DEF OWNER; PURCHASED UNDER QRS TERMS & CONDITIONS	01/01/1999-06/01/1999	LINK TO COMPETITIVE SHOOT; LINK TO UPDATE SHOOT	(USAGE INFORMATION BY USER AND DATA UNIT)	QUALITY INFORMATION	(LINK TO DATA SET)

**Fig. 2**

With reference now to FIG. 2 there is shown geophysical database 32 including, for purposes of illustration, two data records 33A, 33B, each having eight data fields 34A-34H. Data field 34A stores a data shoot identifier by which a particular data shoot can be identified. Data field 34B stores information about the data including a geographical location of the data shoot and description of the data contained in the particular linked data set, including for example shoot parameters such as type of data, size of the area, number of lines, identities of particular data sectors contained within the shoot, etc. Data field 34C stores the identity of the provider of the particular data described in the particular record as well as the payment terms relevant to that data provider. Data field 34D stores the date or date range that the data shoot was collected and may optionally include the date on which the data was entered into geophysical database 32, while data field 34E stores any link(s) to related data.

*Rao*, Fig. 2; ¶ 51 (emphasis added). In other words, *Rao* discloses setting different prices for data records 33A, 33B based on information stored in a specific data field (i.e., data description 34B) of the data record. For example, a record storing marine seismic data (e.g., storing “MARINE AREA” for the data description 34B) may incur a higher price than would a record storing land seismic data (e.g., storing “LAND AREA” for the data description 34B). Respectfully, *Rao* fails to teach setting different prices for the different data fields 34A-34H. For instance, *Rao* does not set any price for the data description field 34B *per se*; *Rao* also does not set any price for the data provider & terms field 34C; and so on. Thus, *Coutts*, even in view of *Rao*, fails to teach or suggest any fee that is specific to a field. In this regard, the Examiner overgeneralization of the term “data type” (i.e., to refer to different *fields*) is inconsistent with the actual meaning

of “data type” as described in *Rao*. Thus, the Examiner has mischaracterized *Rao*. On this basis, the references fail to teach that “each fee schedule for a given logical field defines a fee to be charged when the given logical field is involved in an abstract operation to access a physical entity corresponding to the given logical field.” Accordingly, Applicants respectfully submit that the rejection is defective and should be reversed.

Further, the Examiner mischaracterizes the claims in issue. Specifically, the Examiner asserts as follows:

The Examiner respectfully submits that the terms “field” or “logical field” are not limited to a certain interpretation such as data “columns” of a database record. The Examiner respectfully submits that when interpreting the limitation “each fee schedule for a given logical field defines a fee to be charged...,” one of ordinary skill in the art could certainly have an interpretation that the “logical field” is a label for a unit of data, regardless of the orientation (row or columns) of the data. *Rao* clearly teaches that “pricing structures can include different prices for access to different types of data,” so for accessing a given unit of data, such access would cost a certain amount of price associated with its type.

Advisory Action, page 2 (emphasis added). Respectfully, what is at issue is not the orientation (rows or columns) of the data – but rather, the data that is seen as records and fields. A “field” (also referred to as a “physical field” in Applicants’ Specification) is a term from the computer arts – specifically, from database nomenclature – that persons of ordinary skill in the art will be acquainted with. Persons of ordinary skill in the art will recognize that an entity (of the entity-relationship model (ERM), a well-known database modeling method) refers to a concept from the real world that has *attributes* and that can be modeled in a database. For example, a car has attributes such as make, model, and color. Each attribute of the entity corresponds to a field in the database. Each *instance* of the entity corresponds to a record in the database. For example, the concept of cars can be modeled in the database using three fields: *make*, *model*, and *color*. A particular car, however, can be modeled in the database using a record storing a *value* corresponding to each field – for example, “Ford Focus” (for the *make* field), “ZX4” (for the *model* field) and “Blue” (for the *color* field).

Referring to Figure 2 above, it is irrelevant whether the records 33A, 33B are presented as rows (as shown) or as columns. The issue is whether *Rao* teaches setting

prices that are specific to a field 34A-34H – and *Rao* simply does not. And instead of showing how *Rao* teaches setting prices that are specific to a field, the Examiner merely asserts that *Rao* teaches setting a price for accessing a given “unit of data.” Respectfully, it is not a reasonable interpretation of the claims to equate both the recited physical field and the recited logical field – to a “unit of data.” Those of ordinary skill in the art will recognize that a physical field is not simply a “unit of data.” Instead, a physical field is database terminology that corresponds to attributes (elements 34A-34H in Fig. 2) of an entity being modeled (a data shoot of *Rao*). Further, the claims recite an abstract model that includes a plurality of logical fields and “a mapping rule for each of the plurality of logical fields, which map the plurality of logical fields to physical entities of the data, and wherein each mapping rule comprises an access method which is executed to retrieve a respective physical field of the physical entities of the data.” Respectfully, a logical field that retrieves a corresponding physical field via an associated access method is not simply a “unit of data.” Therefore, the references fail to teach “each fee schedule for a given logical field defines a fee to be charged when the given logical field is involved in an abstract operation to access a physical entity corresponding to the given logical field.” Accordingly, Applicants respectfully submit that the rejection is defective and should be reversed.

**5. Rejection of claims 51-53 are rejected under 35 U.S.C. 103(a) as being unpatentable over *Coutts*, in view of *Rao*, and further in view of *Lyons*.**

*Applicants’ Response to the Examiner’s Argument*

The rejection to claims 51-53 incorporates the rejection to independent claims 46, over *Coutts* in view of *Rao*. As stated above, the rejection to independent claim 46 is believed to be overcome. Accordingly, the rejection to claims 51-53 is also believed to be overcome. Therefore, the reversal of the rejection to these claims is respectfully requested.

## CONCLUSION

The Examiner errs in finding that:

1. Claims 70-74 are directed towards nonstatutory subject matter;
3. Claim 61 is directed towards nonstatutory subject matter;
4. Claims 1-37, 46-50 and 55-74 are unpatentable over *Coutts* in view of *Rao*; and
5. Claims 51-53 are unpatentable over *Coutts* in view of *Rao* and further in view of *Lyons*.

Withdrawal of the rejections and allowance of all claims is respectfully requested.

Respectfully submitted, and  
**S-signed pursuant to 37 CFR 1.4,**

/Gero G. MCCLELLAN, Reg. #44,227/

---

Gero G. McClellan  
Registration No. 44,227  
Patterson & Sheridan, L.L.P.  
3040 Post Oak Blvd. Suite 1500  
Houston, TX 77056  
Telephone: (713) 623-4844  
Facsimile: (713) 623-4846  
Attorney for Appellant(s)

## CLAIMS APPENDIX

1. (Previously Presented) A computer implemented method of providing fee-based access to data, comprising:

providing an abstract model for logically defining abstract operations to access the data, the abstract model comprising:

- (i) a plurality of logical fields;
- (ii) a mapping rule for each of the plurality of logical fields, which map the plurality of logical fields to physical entities of the data, and wherein each mapping rule comprises an access method which is executed to retrieve a respective physical field of the physical entities of the data; and
- (iii) a fee schedule for each of the plurality of logical fields, wherein each fee schedule for a given logical field defines a fee to be charged when the given logical field is involved in an abstract operation to access a physical entity corresponding to the given logical field; and

executing a fee calculator on a computer processor to calculate, based on the fee schedules, a fee to be charged for accessing the data.

2. (Previously Presented) The computer implemented method of claim 1, further comprising:

accessing the data according to an abstract operation comprising at least two of the plurality of logical fields;

wherein the fee to be charged is calculated based on separate fee schedules corresponding to each of the at least two of the plurality of logical fields.

3. (Previously Presented) The computer implemented method of claim 1, further comprising:

providing a run-time component configured with transformation instructions to transform each abstract operation, comprising logical fields selected from the plurality of logical fields, into a physical operation consistent with the physical data.

4. (Previously Presented) The computer implemented method of claim 1, wherein the abstract operation is an abstract query comprising at least two logical fields, and wherein the fee calculator is executed to perform an operation comprising:

accessing a corresponding fee schedule for each of the at least two logical fields;  
determining a per request fee for a first one of the at least two logical fields, wherein the per request fee is charged for each abstract operation involving the first one of the at least two logical fields; and

determining a per item fee for a second one of the at least two logical fields, wherein the per item fee is charged for each instance of the second one of the at least two logical fields involved in a given abstract operation.

5. (Previously Presented) The computer implemented method of claim 4, the operation further comprising multiplying the per item fee by a number of instances of the second one of the at least two logical fields to determine a product, and summing the product and the per request fee to determine the fee to be charged.

6. (Previously Presented) The computer implemented method of claim 1, wherein at least one fee schedule defined by the abstract model specifies a first fee for a first type of operation and a second fee for a second type of operation; and further comprising calculating the fee to be charged based on the type of operation performed.

7. (Previously Presented) The computer implemented method of claim 6, wherein the first type of operation is a query and the second type of operation is one of an insert and an update.

8. (Previously Presented) A computer implemented method of providing fee-based access to physical data comprising a plurality of physical entities each comprising a plurality of physical fields, the method comprising:

providing an abstract model for defining abstract operation specifications logically describing operations to access the data, the abstract model comprising:

(a) a plurality of logical fields;

(b) a mapping rule for each of the plurality of logical fields, which map each of the plurality of logical fields to at least one of the physical entities of the data, and wherein each mapping rule comprises an access method which is executed to retrieve a respective physical field of the physical entities of the data;

(c) a plurality of model entity definitions, each comprising at least one logical field corresponding to a physical field of a physical entity; and

(d) a logical field fee schedule for each of the plurality of logical fields, wherein the fee schedules each specify a fee for accessing a corresponding physical field as part of a physical operation; and

executing a fee calculator on a computer processor to calculate, based on the fee schedules, a fee to be charged for accessing the data.

9. (Previously Presented) The computer implemented method of claim 8, further comprising:

transforming, according to the abstract model, abstract operation specifications into physical operation specifications consistent with the physical data, wherein each abstract operation specification includes at least one of the plurality of model entity definitions.

10. (Previously Presented) The computer implemented method of claim 8, wherein at least one logical field fee schedule defined by the abstract model specifies a first fee for a first type of operation and a second fee for a second type of operation.

11. (Previously Presented) The computer implemented method of claim 10, wherein the first type of operation is a query and the second type of operation is one of an insert and an update.

12. (Previously Presented) A computer implemented method of providing fee-based access to data comprising a plurality of physical entities, each comprising a plurality of physical fields, comprising:

receiving instructions to perform an operation for accessing the data;

performing the operation;  
determining field-specific fees for each of a plurality of the physical fields accessed by the operation; and  
calculating, by operation of a computer processor, a total fee to be charged to a user for the operation.

13. (Previously Presented) The computer implemented method of claim 12, wherein the physical entities are database tables.

14. (Previously Presented) The computer implemented method of claim 12, wherein the operation is one of a query, an insert and an update.

15. (Previously Presented) The computer implemented method of claim 12, wherein determining field-specific fees comprises determining whether a field-specific fee is a per request fee or a per item fee, wherein the per request fee is a singular fee charged for the operation regardless of a number of instances a corresponding physical field is included in the operation and wherein the per item fee is charged for each instance of a corresponding physical field included in the operation.

16. (Previously Presented) The computer implemented method of claim 12, wherein determining field-specific-fees comprises accessing fee schedules for each respective physical field accessed by the operation.

17. (Previously Presented) The computer implemented method of claim 16, wherein each of the fee schedules defines a separate fee for each separate operation type.

18. (Previously Presented) The computer implemented method of claim 17, wherein the separate operation types comprise queries, inserts and updates.



19. (Previously Presented) The computer implemented method of claim 12, wherein determining field-specific fees comprises accessing an abstract model for logically defining the operation accessing the data, the abstract model comprising:

- (i) a plurality of logical fields;
- (ii) a mapping rule for each of the plurality of logical fields, which map the plurality of logical fields to physical entities of the data; and
- (iii) a fee schedule for each of the plurality of logical fields, wherein each fee schedule for a given logical field defines a fee to be charged when the given logical field is involved in an abstract operation to access a physical entity corresponding to the given logical field.

20. (Previously Presented) The computer implemented method of claim 19, wherein each mapping rule comprises an access method for each logical field of an abstract operation specification logically defining the operation accessing the data, and wherein each logical field describes a physical location of a physical entity.

21. (Previously Presented) The computer implemented method of claim 19, wherein each fee schedule defines at least one of a per request fee and a per item fee, wherein the per request fee is a singular fee charged for the operation regardless of a number of instances a corresponding logical field is included in the operation and wherein the per item fee is charged for each instance of a corresponding logical field included in the operation.

22. (Previously Presented) The computer implemented method of claim 19, further comprising:

transforming, according to the abstract model, abstract operations into physical operation consistent with the physical data, wherein each abstract operation includes at least one of the plurality of model entity definitions; and

calculating the fee for executing physical operations based on the fee schedules.

23. (Previously Presented) The computer implemented method of claim 19, further comprising:

accessing the data according to the abstract operation, the abstract operation comprising at least two of the plurality of logical fields; and

calculating the fee to be charged based on separate fee schedules corresponding to each of the at least two plurality of logical fields.

24. (Previously Presented) The computer implemented method of claim 19, further comprising:

providing a run-time component configured with transformation instructions to transform each abstract operation, comprising logical fields selected from the plurality of logical fields, into a physical operation consistent with the physical data; and

providing a fee calculator configured to perform the calculating of the fee for executing physical operations based on the fee schedules.

25. (Previously Presented) The computer implemented method of claim 19, wherein the abstract operation is an abstract query comprising at least two logical fields, the method further comprising:

accessing a corresponding fee schedule for each of the at least two logical fields;

determining a per request fee for a first one of the at least two logical fields, wherein the per request fee is charged for each abstract operation involving the first one of the at least two logical fields; and

determining a per item fee for a second one of the at least two logical fields, wherein the per item fee is charged for each instance of the second one of the at least two logical fields involved in a given abstract operation.

26. (Previously Presented) The computer implemented method of claim 25, further comprising multiplying the per item fee by a number of instances of the second one of the at least two logical fields to determine a product, and summing the product and the per request fee to determine the fee to be charged.

27. (Previously Presented) The computer implemented method of claim 19, wherein at least one fee schedule defined by the abstract model specifies a first fee for a first type of operation and a second fee for a second type of operation; and further comprising calculating the fee to be charged based on the type of operation performed.

28. (Previously Presented) The computer implemented method of claim 27, wherein the first type of operation is a query and the second type of operation is one of an insert and an update.

29. (Previously Presented) A computer-readable storage medium containing a program which, when executed by a processor, performs operations for accessing physical data comprising a plurality of physical entities, each having a plurality of physical fields, the operation comprising:

- receiving instructions to perform an operation accessing the data;
- causing performance of the operation;
- determining field-specific fees for each of a plurality of the physical fields accessed by the operation; and
- calculating a total fee to be charged to a user for the operation.

30. (Previously Presented) The computer-readable storage medium of claim 29, wherein the physical entities are database tables.

31. (Previously Presented) The computer-readable storage medium of claim 29, wherein the operation is one of a query, an insert and an update.

32. (Previously Presented) The computer-readable storage medium of claim 29, wherein determining field-specific fees comprises determining whether a field-specific fee is a per request fee or a per item fee, wherein the per request fee is a singular fee charged for the operation regardless of a number of instances of a corresponding

physical field are included in the operation and wherein the per item fee is charged for each instance of a corresponding physical fee included in the operation.

33. (Previously Presented) The computer-readable storage medium of claim 29, wherein determining field-specific fees comprises accessing fee schedules for each respective physical field accessed by the operation.

34. (Previously Presented) The computer-readable storage medium of claim 33, wherein each of the fee schedules defines a separate fee for each separate operation type.

35. (Previously Presented) The computer-readable storage medium of claim 34, wherein the separate operation types comprise queries, inserts and updates.

36. (Previously Presented) The computer-readable storage medium of claim 29, wherein determining field-specific fees comprises accessing an abstract model for logically defining the operation accessing the data, the abstract model comprising:

- (i) a plurality of logical fields;
- (ii) a mapping rule for each of the plurality of logical fields, which map the plurality of logical fields to physical entities of the data; and
- (iii) a fee schedule for each of the plurality of logical fields, wherein each fee schedule for a given logical field defines a fee to be charged when the given logical field is involved in an abstract operation to access a physical entity corresponding to the given logical field.

37. (Previously Presented) The computer-readable storage medium of claim 36, wherein each mapping rule comprises an access method for each logical field of an abstract operation specification logically defining the operation accessing the data, and wherein each logical field describes a physical location of a physical entity.

38-45. (Canceled)

46. (Previously Presented) A computer implemented method of providing a logical framework for defining abstract operations for accessing physical data comprising a plurality of physical entities each comprising a plurality of physical fields, the method comprising:

providing an abstract model for defining abstract operation specifications logically describing operations to access the data, the abstract model comprising:

- (a) a plurality of logical fields;
- (b) a mapping rule for each of the plurality of logical fields, which map each of the plurality of logical fields to at least one of the physical entities of the data, and wherein each mapping rule comprises an access method which is executed to retrieve a respective physical field of the physical entities of the data;
- (c) a plurality of model entity definitions, each comprising at least one logical field corresponding to a physical field of a physical entity; and
- (d) model entity fee schedules for each of the plurality of model entity definitions, wherein the fee schedules each specify a fee for accessing a physical field of the corresponding model entity definition;

providing a run-time component to transform, according to the abstract model, abstract operation specifications into physical operation specifications consistent with the physical data, wherein each abstract operation specification includes at least one user-selected model entity definitions of the plurality of model entity definitions; and

executing a fee calculator on a computer processor to calculate, based on the fee schedules, a fee to be charged for accessing the data.

47. (Previously Presented) The computer implemented method of claim 46, wherein each of the plurality of physical entities is a table in a database.

48. (Previously Presented) The computer implemented method of claim 46, wherein each physical operation specification is selected from one of an insert statement and an update statement and wherein the model entity fee schedules define different fees for each statement.

49. (Previously Presented) The computer implemented method of claim 46, wherein each physical operation specification is a query, and wherein the model entity fee schedules define fees specific to queries.

50. (Previously Presented) The computer implemented method of claim 46, wherein the abstract operation specification is an abstract query, and further comprising:

- receiving, via a user interface, the abstract query comprising a plurality of query conditions, result fields and a selection of one of the model entity definitions;

- accessing the model entity definition corresponding to the selection;

- determining whether the model entity definition corresponding to the selection specifies one or more required result fields; and

- adding the one or more required result fields to the query upon determining that the model entity definition corresponding to the selection specifies one or more required result fields.

51. (Previously Presented) The computer implemented method of claim 46, further comprising transforming, by the run-time component transforms and according to the abstract model, a single abstract operation specification into at least two separate physical operation specifications consistent with the physical data, wherein each physical operation specification modifies a different physical entity of the data and wherein each physical operation specifications is ordered for execution according to a physical entity relationships specification defining hierarchical relationships between the physical entities of the data.

52. (Previously Presented) The computer implemented method of claim 46, further comprising:

- issuing, by a requesting entity, a request to execute a single abstract operation specification; and

transforming, by the run-time component, the single abstract operation specification into the at least two physical operation specifications for modifying the data.

53. (Previously Presented) The computer implemented method of claim 52, wherein transforming the single abstract operation specification into the at least two physical operation specifications comprises:

generating the at least two physical operation specifications; and  
ordering the at least two physical operation specifications according to a physical entity relationships specification of the abstract model.

54. (Canceled)

55. (Previously Presented) The computer implemented method of claim 46, wherein each logical field describes a location of the physical entities of the data.

56. (Previously Presented) The computer implemented method of claim 46, further comprising a logical field fee schedule for each of the plurality of logical fields, wherein the fee schedules each specify a fee for accessing a corresponding physical field as part of a physical operation specification.

57. (Previously Presented) The computer implemented method of claim 56, wherein at least one fee schedule defined by the abstract model specifies a first fee for a first type of operation and a second fee for a second type of operation.

58. (Previously Presented) The computer implemented method of claim 57, wherein the first type of operation is a query and the second type of operation is one of an insert and an update.

59. (Previously Presented) A computer-readable storage medium containing a program which, when executed by a processor, provides a logical framework for defining abstract query operations, the program comprising:

an abstract model for defining abstract queries logically describing operations to query the data, the abstract model comprising:

- (i) a plurality of logical fields;
- (ii) a mapping rule for each of the plurality of logical fields, which map the plurality of logical fields to physical entities of the data, and wherein each mapping rule comprises an access method which is executed to retrieve a respective physical field of the physical entities of the data; and
- (iii) a fee schedule for each of the plurality of logical fields;

a run-time component configured with transformation instructions to transform an abstract query, comprising logical fields selected from the plurality of logical fields, into a physical query consistent with the physical data; and

a fee calculator configured to calculate a fee for executing physical queries based on the fee schedules.

60. (Previously Presented) The computer-readable storage medium of claim 59, wherein the mapping rules comprise an access method for each of the plurality of logical fields, wherein each logical field describes a location of the physical entities of the data.

61. (Previously Presented) A computer comprising:

a memory;

at least one computer processor; and

a logical framework for defining abstract modification operations for modifying physical data, the logical framework comprising:

an abstract model for defining an abstract modification specification logically describing an operation to modify the data, the abstract model comprising:

- (i) a plurality of logical fields;



(ii) a mapping rule for each of the plurality of logical fields, which map the plurality of logical fields to physical entities of the data, and wherein each mapping rule comprises an access method which is executed to retrieve a respective physical field of the physical entities of the data; and

(iii) a fee schedule for each of the plurality of logical fields;

a run-time component to transform an abstract query, comprising logical fields selected from the plurality of logical fields, into a physical query consistent with the physical data; and

a fee calculator which when executed configures the at least one computer processor to calculate a fee to be charged for executing physical queries based on the fee schedules.

62. (Original) The computer of claim 61, wherein each fee schedule defines at least one of a per request fee and a per item fee, wherein the per request fee is a singular fee charged for the operation regardless of a number of instances a corresponding logical field is included in the operation and wherein the per item fee is charged for each instance of a corresponding logical field included in the operation.

63. (Previously Presented) The computer of claim 61, wherein each logical field describes a location of the physical entities of the data.

64. (Previously Presented) A computer implemented method of providing fee-based access to data comprising a plurality of physical entities, each comprising a plurality of physical fields, comprising:

receiving, via a user interface, user input comprising instructions for an operation for accessing the data selected fields of the plurality of the physical fields;

determining field-specific fees for each of the selected fields;

calculating, by a computer processor, a fee to be charged to a user for accessing the selected fields; and

displaying the fee to the user via a user interface.

65. (Previously Presented) The computer implemented method of claim 64, wherein the physical entities are database tables.

66. (Previously Presented) The computer implemented method of claim 64, wherein determining field-specific fees comprises determining whether a field-specific fee is a per request fee or a per item fee, wherein the per request fee is a singular fee charged for the operation regardless of a number of instances a corresponding physical field is included in the operation and wherein the per item fee is charged for each instance of a corresponding physical field included in the operation.

67. (Previously Presented) The computer implemented method of claim 64, wherein determining field-specific fees comprises accessing fee schedules for each respective physical field accessed by the operation.

68. (Previously Presented) The computer implemented method of claim 67, wherein each of the fee schedules defines a separate fee for each separate operation type.

69. (Previously Presented) The computer implemented method of claim 68, wherein the separate operation types comprise queries, inserts and updates.

70. (Previously Presented) A computer implemented method for displaying fee information for fee-based access to data comprising a plurality of physical entities, each comprising a plurality of physical fields, comprising:

- displaying one or more user interface screens for construction of queries;
- receiving, via the one or more user interface screens, user input defining a query configured to access selected fields of the plurality of physical fields; and
- displaying, via the one or more user interface screens on an output device, a field-specific access fee for each of the selected fields.

71. (Previously Presented) The computer implemented method of claim 70, further comprising displaying a per query fee, the per query fee being a singular fee charged for the query regardless of a number of instances a corresponding physical field is included in the query.

72. (Previously Presented) The computer implemented method of claim 70, further comprising determining the field-specific fees by accessing an abstract model for logically defining the query, the abstract model comprising:

- (i) a plurality of logical fields;
- (ii) a mapping rule for each of the plurality of logical fields, which map the plurality of logical fields to physical entities of the data; and
- (iii) a fee schedule for each of the plurality of logical fields, wherein each fee schedule for a given logical field defines a fee to be charged when the given logical field is involved in a query to access a physical entity corresponding to the given logical field.

73. (Previously Presented) The computer implemented method of claim 72, wherein each mapping rule comprises an access method for each logical field of an abstract operation specification logically defining the operation accessing the data, and wherein each logical field describes a physical location of a physical entity.

74. (Previously Presented) The computer implemented method of claim 72, wherein each fee schedule defines at least one of a per request fee and a per item fee, wherein the per request fee is a singular fee charged for the query regardless of a number of instances a corresponding logical field is included in the query and wherein the per item fee is charged for each instance of a corresponding logical field included in the query.

## EVIDENCE APPENDIX

None.

## RELATED PROCEEDINGS APPENDIX

None.